JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 25 No. 12, pp. 165-172, December 2020
https://doi.org/10.9708/jksci.2020.25.12.165

Framework for Efficient Web Page Prediction using Deep Learning

Kyung-Chang Kim*

*Professor, Dept. of Computer Engineering, Hongik University, Seoul, Korea

[Abstract]

Recently, due to exponential growth of access information on the web, the importance of predicting a user's next web page use has been increasing. One of the methods that can be used for predicting user's next web page is deep learning. To predict next web page, web logs are analyzed by data preprocessing and then a user's next web page is predicted on the output of the analyzed web logs using a deep learning algorithm. In this paper, we propose a framework for web page prediction that includes methods for web log preprocessing followed by deep learning techniques for web prediction. To increase the speed of preprocessing of large web log, a Hadoop based MapReduce programming model is used. In addition, we present a web prediction system that uses an efficient deep learning technique on the output of web log preprocessing for training and prediction. Through experiment, we show the performance improvement of our proposed method over traditional methods. We also show the accuracy of our prediction.

▸ **Key words**: Deep learning, Framework, Web page prediction, Web log, Log preprocessing, MapReduce model

[요 약]

웹에서 접근하는 정보의 폭발적인 증가에 따라 사용자의 다음 웹 페이지 사용을 예측하는 문제의 중요성이 증가되었다. 사용자의 다음 웹 페이지 접근을 예측하는 방법 중 하나가 딥 러닝 기법이다. 웹 페이지 예측 절차는 데이터 전처리 과정을 통해 웹 로그 정보들을 분석하고 딥 러닝 기법을 이용하여 분석된 웹 로그 결과를 가지고 사용자가 접근할 다음 웹 페이지를 예측한다. 본 논문에서는 웹 페이지 예측을 위한 효율적인 웹 로그 전처리 작업과 분석을 위해 딥 러닝 기법을 사용하는 웹 페이지 예측 프레임워크를 제안한다. 대용량 웹 로그 정보의 전처리 작업 속도를 높이기 위하여 Hadoop 기반 맵/리듀스(MapReduce) 프로그래밍 모델을 사용한다. 또한 웹 로그 정보의 전처리 결과를 이용한 학습과 예측을 위한 딥 러닝 기반 웹 예측 시스템을 제안한다. 실험을 통해 논문에서 제안한 방법이 기존의 방법과 비교하여 성능 개선이 있다는 사실을 보였고 아울러 다음 페이지 예측의 정확성을 보였다.

▸ **주제어**: 딥러닝, 프레임워크, 웹 페이지 예측, 웹 로그, 로그 전처리, 맵/리듀스 모델

• First Author: Kyung-Chang Kim, Corresponding Author: Kyung-Chang Kim
*Kyung-Chang Kim (kckim@hongik.ac.kr), Dept. of Computer Engineering, Hongik University
• Received: 2020. 12. 04, Revised: 2020. 12. 23, Accepted: 2020. 12. 23.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

큰 폭의 빈번한 웹 사용은 서버 부하와 접근 지연과 같은 결과를 가져온다. 이런 문제들을 해결하기 위한 방법 중 하나가 웹 페이지의 사전 검색이다. 웹 사전 검색은 다음 웹 페이지 접근을 예측하는 하나의 방법으로 접근 지연과 서버 부하를 줄이기 위하여 사용자가 요청하기 전에 사용자가 접근할 다음 웹 페이지를 클라이언트의 캐쉬(cache)로 미리 가져오는 것이다.

다음 웹 페이지 접근을 예측하는 또 다른 방법은 웹 사용 마이닝(mining)이다. 본 논문에서 제안한 프레임워크도 웹 페이지 접근 예측을 위해 웹 사용 마이닝 기법에 기반 한다. 웹 사용 마이닝은 웹 로그를 마이닝 하여 유용한 패턴과 지식을 찾는 방법이다[1,2]. 웹 사용 마이닝은 데이터 전처리, 패턴 발견, 그리고 패턴 분석의 3 단계로 이루어진다[3]. 데이터 전처리 단계가 가장 중요한데 그 이유는 전체 처리 과정에서 가장 시간이 많이 걸릴 뿐 아니라 그 출력이 다음 단계들의 결과를 좌우하기 때문이다. 일부 연구가들은 전체 과정에서 데이터 전처리 작업이 80%를 차지한다고 얘기하고 있다.

데이터 전처리 단계의 주요 작업들은 데이터 정제, 사용자 식별 그리고 세션(session) 식별이며 그중에서 세션 식별이 가장 어려운 작업이다. 세션의 정의는 사용자가 어떤 웹 사이트를 방문 시 요청한 웹 페이지의 순서이다. 어떤 기간 동안 사용자는 하나 혹은 여러 세션을 가질 수 있다.

본 논문에서는 대규모 웹 로그 정보의 전처리 작업 속도를 높이기 위해 전처리 일부 작업들을 합병하고 병렬 처리를 위한 MapReduce 프로그래밍 모델을 사용하고자 한다[4]. MapReduce 알고리즘은 대규모 데이터 집합을 처리하고 생성하는데 많이 사용되는 모델이다.

데이터 전처리 작업에 의해서 세션이 생성되면 그 세션을 이용하여 사용자의 다음 웹페이지 접근을 예측하는데 딥 러닝 기법들이 이용될 수 있다[5]. 본 논문에서는 딥 러닝 기법 중 Long Short-Term Memory (LSTM) 망을 적용하고자 한다. LSTM 망은 RNN 모델[6] 중 하나로 망의 노드들을 directed cyclic graph (DCG)로 연결한다. LSTM 망은 시계열 데이터의 분류와 처리에 보다 특화 되었다고 볼 수 있다. 다음 웹 페이지 예측을 위한 웹 로그 정보들도 일종의 시계열 데이터하고 볼 수 있다.

본 논문의 기여는 다음과 같다. 첫 번째, 대규모 웹 로그 정보의 전처리 작업 속도를 높이기 위하여 MapReduce 프로그래밍 모델을 채택하였다. 두 번째, 다음 방문할 웹 페이지를 예측하기 위하여 시계열 데이터 분석에 적합한 RNN 딥 러닝 알고리즘의 하나인 LSTM을 사용하였다.

실험을 통하여 본 논문에서 제안한 프레임워크가 기존의 방법론과 비교하여 대규모 웹 로그 정보를 처리하는데 성능이 보다 우수하다는 것을 입증하였고 예측의 정확성도 높였다는 것을 입증하였다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구들을 검토하고 3장에서는 제안한 데이터 전처리 기법을 소개하고 다음 웹페이지 방문을 예측하는 프레임워크를 설명한다. 또한 실험 환경과 결과를 보여주고 마지막으로 4장에서는 결론으로 맺는다.

II. Preliminaries

1. Related works

웹 마이닝의 데이터 전처리 단계에서 가장 어려운 작업이 세션 식별이다. 세션 식별에는 시간 기반, 항해 기반 그리고 이들을 혼합하는 하이브리드 방법이 있는데 논문에서는 시간 기발 세션 식별에 중점을 두고 있다.

Zhou[7]에서는 세션 식별을 위해서 고정 시간 threshold 값을 사용하였으며 기본 시간 종료 threshold 값으로 30분을 설정하여 사용하였다. Castellano[8] 에서는 LODAP라는 로그 데이터 전처리 도구를 개발하여 웹 사이트의 로그 파일에 저장된 웹 요청 정보들로부터 사용자 세션을 추출하였다. 세션을 식별하기 위해서 고정 시간 기반 방법이 사용되었다.

Peng[9] 에서는 평균 threshold 값에 기반 한 세션 식별 알고리즘을 제안하였다. 사용자들이 특정 페이지에 머무르는 시간이 전부 다르기 때문에 고정 threshold 값으로 긴 세션을 여러 세션으로 잘못 나눌 수 있다. 이 문제를 해결하기 위해 저자들은 고정 값 대신 평균 threshold 값을 제안하였다. 첫 단계에서는 평균 threshold 값을 계산하여 세션을 식별하는데 사용하였고 두 번째 단계에서는 오류를 제거하기 위해서 전에 식별한 세션을 다시 식별하였다. 실험 결과를 통해 긴 세션에 대해서 세션 식별의 정확성을 보였다.

Xinhua[10]는 동적 시간 종료에 기반 한 세션 식별 알고리즘을 제안하였다. 알고리즘의 첫 번째 부분에서는 페이지의 중요도와 통계 결과를 결합하여 각 웹 페이지의 시간 종료를 결정하였다. 알고리즘의 나머지 부분에서는 시간 종료를 동적으로 조절하였다. 실험 결과 기존의 시간 기반 알고리즘과 비교하여 제안한 알고리즘이 보다 우수한 성능을 보였다. Matrix를 구축하여 세션을 식별하는 새로운 기법이 제안되었다[11]. Matrix의 열(column)은 웹

페이지를 나타냈고 행(row)은 사용자와 그들의 세션을 나타냈다. 브라우징 시간과 무게(weight)는 사용자의 항해 데이터를 이용하여 계산하였다. 이들은 matrix cell에 저장되었고 각 행은 사용자의 개인 세션 항해로 인식되었다. 만일 무게가 100이면 다음 값은 새로운 세션으로 다음 행에 저장되었다.

사용자의 다음 웹 페이지 방문을 예측하기 위해서 최근에는 인공 신경망(artificial neural network, ANN)을 사용하였다. ANN을 이용한 몇 개의 연구들을 소개하고자 한다. Om[5]에서는 feed forward ANN과 k-mean clustering 방법을 이용하여 예측 모델을 제안하였다. 저자들은 현재 사용자의 세션에서 가장 가까운 사용자 세션 클러스터(cluster)를 찾기 위해서 ANN을 이용하였고 사용자 세션들을 흡사도(similarity)에 의한 클러스터링을 하기 위해서 k-mean clustering을 이용하였다. 이들 클러스터에 대해서 feed forward ANN 모델을 학습하여 현재 사용자 시퀀스와 가장 높은 클러스터를 찾는데 사용되었다.

Pruthvi[12] 모델에서는 대규모 데이터 집합을 신속하게 학습하기 위해서 ANN을 Hadoop 프레임워크에 있는 map-reduce 프로그래밍 모델로 구현하였다. Vidushi[13] 제안 모델에서는 ANN과 self-organizing map (SOM)을 사용하였다. 이 모델에는 3가지 층(layer)으로 이루어 졌는데 이들 층들은 클러스터링, SOM 그리고 ANN이다. 클러스터링 층에서는 클러스터링 방법을 이용하여 데이터 세트를 filtering 하였다. 방문 횟수가 높은 페이지들을 나타내는 클러스터를 추가 처리를 위해 선택하였다. 다음 SOM 층에서는 여러 웹 페이지에 weight를 부여하기 위하여 SOM을 적용하여 웹 페이지 사용과 예측을 분석한다. 마지막으로 ANN 층에서는 다음 웹 페이지 예측을 수행하기 위해 ANN 모델을 학습시킨다.

대부분의 연구에서는 웹 페이지 예측을 위해 ANN 모델과 feed forward 신경망을 이용하였다. 본 논문에서는 이들과 달리 웹 페이지 예측을 위해 recurrent neural network (RNN) 모델을 이용하고자 한다.

III. The Proposed Scheme

1. Framework for Web Prediction System

이장에서는 제안한 웹 마이닝과 딥 러닝 기법을 이용한 웹 페이지 예측 프레임워크에 대해서 기술한다. 제안한 웹 페이지 예측 시스템은 그림 1에서 보듯이 3개의 주요 컴포넌트인 데이터 전처리(data preprocessing), 데이터 모델링(data modeling) 그리고 데이터 예측(prediction)으로 구성된다.

예측을 위한 사용자 세션을 모델링하기 전에 대량의 원시 로그 파일들이 분석되고 사용자가 방문한 웹 페이지들의 순서(즉 세션)가 형성되어야 한다. 로그 파일의 레코드들은 특정 형식(format)을 갖는데 일반적인 형식인 common log format (CLF)은 IP 주소, 데이터 요청 시간, 요청 방법, 요청된 파일(웹 페이지), HTTP 버전, 오류 상태 그리고 전송된 바이트 수를 포함한다. 본 제안에서도 CLF 형식(그림 2)을 갖는 서버 접근 로그를 다룬다.

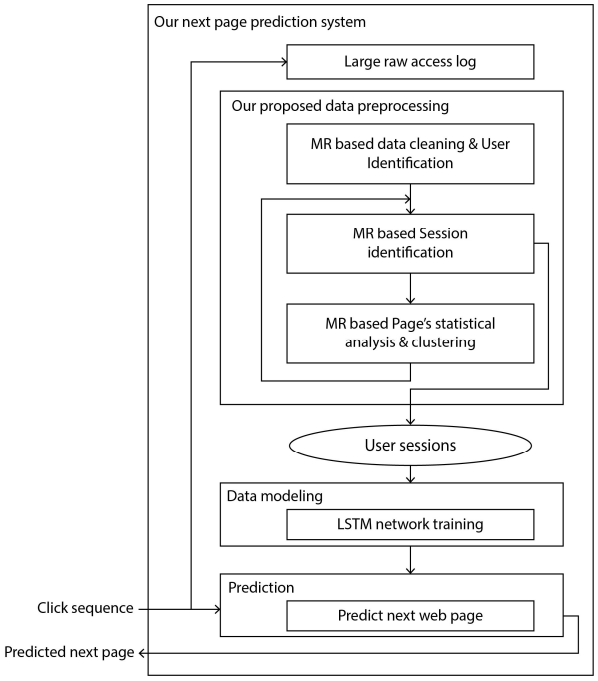

Fig. 1. Framework of Web Prediction System

데이터 전처리 컴포넌트는 데이터 정제 및 사용자 식별 단계, 세션 식별 단계, 페이지 통계 분석 및 클러스터링 단계, 그리고 재식별 단계로 이루어진다. 데이터 정제 및 사용자 식별 단계에서는 웹 접근 로그를 분석하여 필요 없는 레코드(즉 gif, jpg, css등)들을 제거한다. 또한 오류 상태 요청, 아주 흔하거나 드문 레코드들을 제거한다. 여기서 아주 드문 페이지란 엔트리가 오직 하나인 세션을 말한다. 가끔 어떤 웹 사용자가 특정 웹 사이트 방문 후 1단계 후 바로 그 사이트를 나가는 경우가 있다. 그러한 방문은 웹 사이트 사용과 관련하여 의미 없는 지식이다. 아주 흔한 페이지란 index.html이나 로그 페이지와 같은 웹 사이트에 대한 엔트리 페이지이다. 사용자 식별에서는 사용자 순서가 같은 IP 주소, 브라우저 타입 그리고 운영 체제 타입으로 식별된다.

세션 식별자 단계에서는 사용자 세션이 식별된다. 식별은 특정 페이지의 순차적인 시간차 요청이 30분의 정적 threshold 값을 넘었을 때 결정된다.

페이지 통계 분석 및 클러스터링 단계에서는 페이지 사용 통계는 페이지 사용자 세션을 분석하여 얻어진다. 페이지들은 페이지 사용 통계를 기반으로 k-means 클러스터링 기법을 이용하여 k 클러스터로 그룹핑(grouping) 된다. 각 클러스터의 threshold 값은 해당 페이지들의 평균 머무른 시간으로 계산된다.

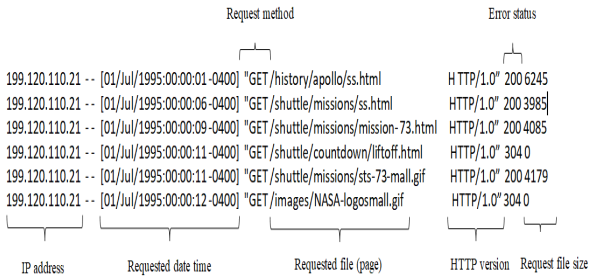

Fig. 2. CLF format of Web Log

재식별 단계에서는 순차적인 페이지 요청 시간이 페이지 클러스터의 threshold 값을 넘느냐에 따라서 사용자 세션을 재식별 한다. 그림 3은 데이터 전처리 단계들이 처리 된 후의 사용자 세션 결과를 보여준다.

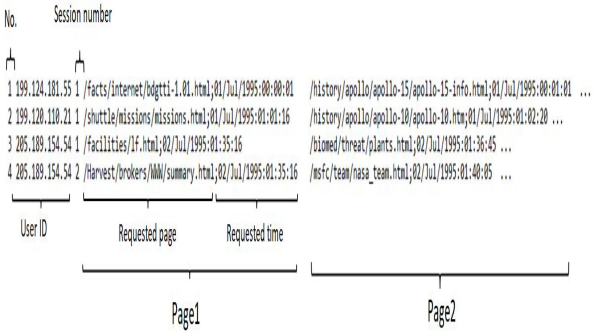

Fig. 3. Result after Preprocessing Step

본 논문에서 제안하는 데이터 전처리 컴포넌트의 특징은 전처리 시간을 줄이기 위하여 모든 단계에서 MapReduce (MR)알고리즘을 적용하였다.

데이터 모델링 컴포넌트에서는 웹 접근 로그로부터 사용자 세션이 얻어진 후 다음 웹 페이지 예측을 위해 세션들을 모델링 한다. 모델링을 위해 3개의 숨은 레이어(hidden layer)를 갖는 LSTM 망을 사용한다. LSTM 망은 recurrent 신경망(RNN)의 일종으로 노드들의 연결을 directed cyclic graph (DCG)로 모델링 한다.

또 다른 모델은 feed-forward 신경망(FNN)으로 노드들의 연결은 입력 노드로부터 숨은 노드를 거쳐 출력 노드로 한쪽 방향으로 연결된다. FNN에서는 노드들 사이에 cycle 연결이 없다(그림 4 참조). DNN은 FNN의 변형으로 입력과 출력 노드 사이에 많은 숨은 레이어로 구성된다.

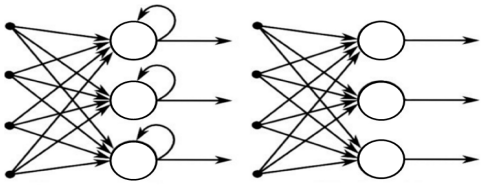

Fig. 4. Comparison of RNN(left) and FNN(right)

본 논문에서는 제안한 LSTM에 기반한 예측 모델과 기존의 deep 신경망(DNN)과 성능 비교를 실시하여 제안한 예측 모델의 효율성을 보이고자 한다. LSTM 망은 LSTM 단위로 구성되는데 하나의 LSTM 단위는 보통 셀(cell), forget gate, 입력 gate, 출력 gate로 구성된다 (그림 5). 셀은 입력을 받아 일정 시간 저장한다. 일반적으로 forget gate는 어떤 값이 다음 단계로 보내지고 어떤 값이 셀에 남는지를 제어한다. 입력 gate는 어떤 새로운 입력이 셀에 저장되는지 제어하고 출력 gate는 어떤 정보가 출력되는지 결정한다.

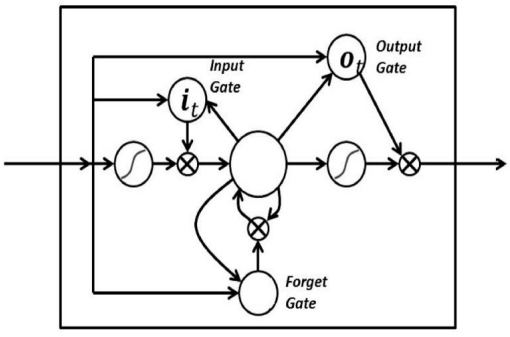

Fig. 5. Sample LSTM unit

LSTM 망은 분류하고 처리하는데 있어 시계열 데이터가 더 적합하다. 본 논문에서는 시퀀스의 마지막 웹 페이지를 출력으로 통과하고 나머지 웹 페이지들은 입력으로 통과한다. 예를 들면, 만일 사용자 세션의 길이가 k 이면, 세션의 k-1 페이지들을 입력 노드들로 통과시키고 k 번째 페이지는 출력 노드로 통과시킨다.

그림 6은 다음 페이지 예측을 위한 LSTM 망의 구조를 보여준다. 학습 분류를 하는 동안 사용자 세션의 웹 페이지들이 충분한 학습 후에 사용자 세션의 마지막 웹 페이지로 분류된다. 이 단계의 출력이 학습 모델이고 이 모델이 다음 단계에 사용된다.

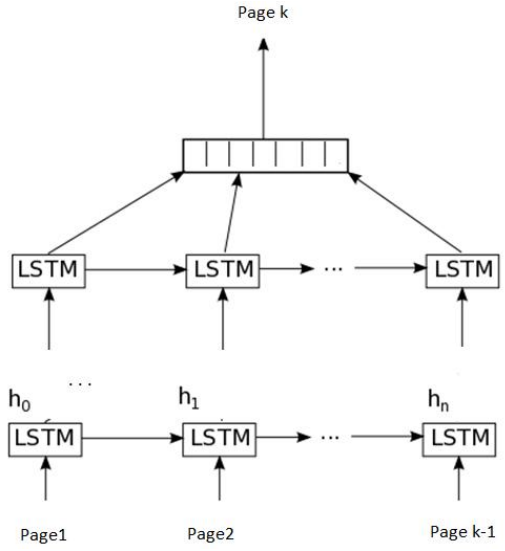

Fig. 6. Sample structure of LSTM network for next web page prediction

예측 컴포넌트에서는 사용자가 웹 페이지 예측 시스템으로 클릭 시퀀스를 보낸다. 그러면 사용자가 현재 방문한 페이지를 로그 파일에 기록시키고 사용자의 클릭 시퀀스를 예측 모델로 보낸다. 이 예측 모델은 숨겨진 3개의 layer가 있는 LSTM 망이다. 예측 모델이 다음 웹 페이지를 예측할 때 기존의 학습된 모델을 사용한다. 클릭 시퀀스의 페이지들이 예측 모델로 보내지면 사용자가 다음 방문할 확률이 높은 웹 페이지가 출력으로 예측된다. 그러면 예측 시스템은 현재 사용자의 예측된 다음 페이지를 출력한다.

2. Experimental Result

몇 가지 실험을 통하여 제안된 프레임워크의 효율성을 입증하고자 한다. 첫 번째는 데이터 전처리 단계에 대한 효율성 실험을 실시하였고 두 번째 실험에서는 다음 웹 페이지 예측에 대한 정확성 실험을 실시하였다.

2.1. Experimental Environment

실험에 사용된 시스템은 Intel(R) Core(TM) i7-3370이고 CPU는 3.5 GHz, GTX 1070Ti와 11GB RAM을 사용하였고 운영체제는 Ubuntu 18.04 LTS이다. 데이터 전처리 단계에서는 Hadoop을 구현하였고 다음 페이지 예측을 구현하기 위한 학습 platform으로 Tensorflow 라이브러리를 사용하였다.

2.2. Data Preprocessing Experiment

먼저 데이터 전처리 단계의 세션 식별에 대한 효율성을 실험하기 위하여 본 논문에서 제안한 알고리즘과 정적 threshold 값을 갖는 기존의 시간 기반 알고리즘의 성능을 비교하였다. 본 실험에서는 threshold 값을 11분으로 하였다.

이 실험을 위하여 전체 98,400 레코드를 갖는 특정 사이트의 로그를 사용하였다. 제안한 알고리즘에서는 데이터 정제와 사용자 식별 단계를 거친 후 12,403 레코드를 얻을 수 있었다. 수작업 식별을 통하여 1874개의 실제 세션이 있음을 측정하였다. 표 1은 제안한 알고리즘과 기존의 알고리즘에서 식별한 세션의 개수와 실제 세션의 개수를 보여주고 있다.

Table 1. Result of session identification

	Number of session identified	Number of real session identified
Traditional time-based session identification (with 11 minutes static threshold)	1381	1014
The proposed technique	1415	1153

표 2는 비교한 두 알고리즘의 효과율과 식별 율을 보여주고 있다. 효과율(effective rate)은 식별한 실제 세션과 모든 세션의 비(ratio)를 의미하고 식별 율(identification rate)은 식별한 실제 세션과 전체 실제 세션의 비를 의미한다. 표 2에서 보듯이 본 논문에서 제안한 알고리즘이 기존의 알고리즘보다 더 효과적으로 사용자 세션을 식별한다는 것을 알 수 있다.

Table 2. Comparison of effective and identification rate

	Effective rate	Identification rate
Traditional technique	73%	54%
Proposed technique	81%	61%

다음 실험은 데이터 전처리 단계의 실행 시간을 측정하였다. 성능 평가는 제안한 Hadoop MapReduce(MR)에

기반 한 알고리즘과 Hadoop에 기반 하지 않은 기존의 알고리즘을 사용했을 경우를 비교하였다.

이 실험은 NASA 데이터 세트[14]를 이용하였다. 이 데이터 세트는 NASA Kennedy 우주 센터에서 수집한 1개월 분 웹 요청으로 로그에 전체 약 1,600,000 레코드를 포함한다. 전처리 단계에서 데이터 정제 후 약 200,000 레코드를 얻었다. 비교한 기존의 알고리즘은 능동 사용자 기반 식별 알고리즘과 일반적인 식별 알고리즘이다.

표 3은 실행 시간과 식별한 세션의 개수에 대한 성능 평가 결과를 보여주고 있다. 제안한 Hadoop MapReduce 기반 알고리즘이 기존의 알고리즘에 비하여 전처리 실행 시간이 월등히 빠를 뿐만 아니라 더 많은 세션을 식별한 것을 알 수 있다. 물론, 제안한 알고리즘이 데이터 전처리에 있어 성능의 우수성을 객관적으로 보이기 위해서는 보다 다양한 데이터 세트로 실험을 실시할 필요가 있다.

Table 3. Comparison of execution time

	Execution time (Sec)	No. of sessions identified
ActiveUser identification algorithm	2038	79589
Trivial user identification algorithm	4255	79589
The proposed session identification	79	111928

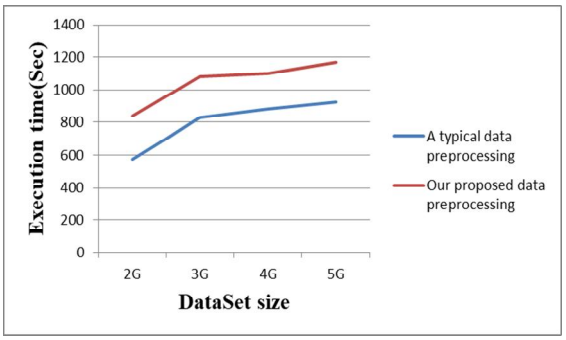

Fig. 7. Execution time of MR based data preprocessing on large log data set

추가적으로 제안한 알고리즘과 MapReduce에 기반한 기존의 데이터 전처리 기법과 비교하였다. 2GB, 3GB, 4GB 그리고 5GB 대규모 로그 데이터 세트에 대해서 실험하였다. 그림 7은 단일 노드 클러스터를 이용하여 처리를 완료하는 시간을 보여준다.

제안한 알고리즘을 이용 시 데이터 전처리 실행 시간이 늘어난다. 그 이유는 기존의 기법과 달리 보다 많은 단계를 포함하기 때문이다. 이러한 단점에도 불구하고 추가 단계들을 적용하면 보다 많은 효율적인 세션을 생성할 수 있고 예측의 정확성을 높일 수 있다는 장점이 있다.

2.3. Next Page Prediction Experiment

웹 페이지 예측의 정확성을 측정하기 위하여 대규모 NASA 와 ClarkNet[15] 데이터 세트를 이용하여 실험을 진행하였다. ClarkNet 데이터 세트는 ClarkNet 서버에 대한 1 주일분의 모든 데이터 요청으로 총 1,673, 711 레코드가 포함된다.

웹 페이지 예측에 대한 실험은 제안한 전처리 기법을 이용하여 식별한 사용자 세션에 기반 하였다. 본 실험에서는 DNN과 LSTM의 두 가지 딥러닝 모델을 사용하였다. 첫 번째 실험에서는 길이가 10 페이지인 사용자 세션을 사용하여 두 모델을 학습하였다. 그림 8은 두 예측 모델에 대한 결과를 보여주고 있다. NASA 와 ClarkNet 데이터 세트에 대해서 DNN 모델을 사용하여 각각 81%와 84% 정확성을 얻었으며 LSTM 모델을 사용 시 각각 95%와 98% 정확성을 얻었다.

두 번째 실험에서는 LSTM 모델을 이용하여 세션 길이를 6에서 10까지 변화하면서 정확성 실험을 실시하였다. 그림 9는 예측 정확성과 사용자 세션의 길이에 대한 관계를 보여주고 있다. 세션의 길이를 6에서 10 페이지로 증가했을 때 예측의 정확성도 덩달아 높아졌다.

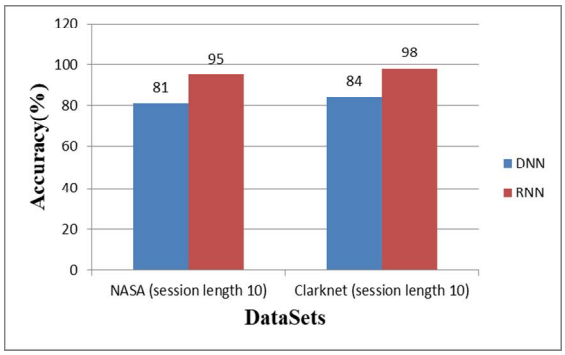

Fig. 8. Comparison of accuracy of prediction models

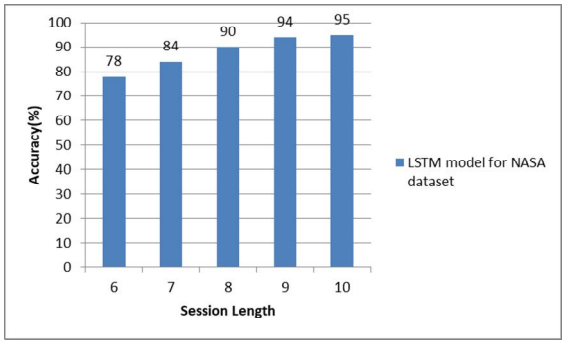

Fig. 9. The effect of session length on accuracy

IV. Conclusions

본 논문에서는 MapReduce 알고리즘과 딥러닝 알고리즘에 기반한 다음 웹 페이지 예측 시스템에 대한 프레임워크를 제안하였다. 대규모로 쌓인 웹 로그 결과는 보다 정확한 예측을 가능하게 하지만 전처리 시간이 늘어나는 부작용이 생긴다. 이 문제를 해결하기 위하여 도든 전처리 단계에서 MapReduce 모델에 기반하여 Hadoop에 구현하였다. 또한 보다 정확한 시간 기반 세션 식별을 위하여 k cluster threshold 값을 사용하는 기법을 이용하였다.

데이터 전처리에 대한 실험에서 본 논문에서 제안한 k cluster threshold 값을 사용한 시간 기반 세션 식별이 기존의 시간 식별 방법에 비하여 보다 효율적인 세션을 생성하는 것을 보였다. 또한 제안한 MapReduce 모델 기반 전처리 알고리즘들을 Hadoop에서 구현하였을 때 대규모 로그 전처리 실행 시간을 대폭 줄여지는 것을 보였다.

기존의 전처리 기법들에 비해서 제안 기법에는 추가적으로 페이지 통계 분석, 클러스터링 그리고 재식별 단계가 포함되지만 시간을 줄이기 위하여 일부 단계들을 통합하였고 Hadoop MapReduce 모델로 구현하였다. 실험을 통하여 기존의 기법에 비하여 전처리 시간은 줄어들었고 효율적인 세션이 생성되었음을 보였다.

보다 정확한 다음 웹 페이지 예측을 위하여 본 논문에서는 딥러닝 알고리즘을 사용하였다. 실험 결과 제안한 LSTM 네트워크 모델이 일반적인 DNN 네트워크 모델에 비하여 예측의 정확성이 높여졌음을 보였다. 추후 연구로 다른 공개된 데이터 집합(세트)에 대해서 K-클러스터 임계치에 대한 평가를 고려해 볼 수 있다.

REFERENCES

[1] Neha Sharma, Pawan Makhija “Web usage Mining: Web user Session Construction using Map-Reduce”, Global journal of Computer Science and Technology (E), volume 17, issue 4, 2017.

[2] Zidrina Pabarskaite, Aistis Raudys, “A process of knowledge discovery from web log data: Systemization and critical review”, Journal of Intelligent Information System, Springer, 2007.

[3] Natheer Khasawnech, Chien-Chung Chan. “Active User-Based and Ontology-based Log Data Preprocessing for Web Usage Mininig” Proceedings of th 2006 ACM international conference on web Intelligence Applications, 2006

[4] Jeffrey Dean and Sanjay Ghemawat, “MapReduce: Simplifed Data Processing on Large Clusters” OSDI 2004

[5] Om Prakash Mandal, Hiteshware Kumar Azad “Web Access Prediction Model using Clustering and Artificial Neural Network”, IJERT, Vol.3 Issue 9, 2014

[6] Https://towardsdatascience.com/recurrent-neural-networks-and-lstm-4b601dd822a

[7] Zhou, B., Hui, S. and Fong, A. “An effective approach for periodic web personalization”, 2006 IEEE/WIC/ACM International Conference on Web Intelligence, 2006, pp. 284-292.”, Journal of Intelligent Information System, Springer, 2007

[8] Castellano, G., Fanelli, A.M. and Torsello, M.A. (2007), “LODAP: a log data preprocessor for mining web browsing patterns”, Proceedings of the 6th Conference on 6th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases, World Scientific and Engineering Academy and Society (WSEAS), 2007, pp. 12-17

[9] Peng, Z. and Zhao, M. “Session identification algorithm for web log mining”, 2010 International Conference on Management and Service Science, IEEE, 2010 pp. 1-4.

[10] Xinhua, H. and Qiong, W. , “Dynamic timeout-based a session identification algorithm”, 2011 International Conference on Electric Information and Control Engineering, IEEE, 2011, pp. 346-349

[11] Chitraa, V. and Thanamani, A., “A novel technique for sessions identification in web usage mining preprocessing”, International Journal of Computer Applications, Vol. 34 No. 9, 2011, pp. 24-28.

[12] Pruthvi, “Web-Users’ Browsing behavior Prediction by Implementing Neural Network in MapReduce”, IJAFRC, Vol.1 Issue 5, 2014

[13] Vidushi, Yashpal Singh, “SOM Improved Neural Network Approach for Next Page Prediction”, International Journal of Computer Science and Mobile Computing, Vol.4 Issue.5, May-2015, pg. 175-181

[14] Http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html

[15] Http://ita.ee.lbl.gov/html/contrib/ClarkNet-HTTP.html

Authors

Kyung-Chang Kim received the B.S. in Computer Science from Hongik University in 1978, M.S. in Computer Science from KAIST in 1980 and Ph.D. in Computer Science from University of Texas at Austin in 1990.

Dr. Kim joined the faculty of the Department of Computer Engineering at Hongik University, Seoul, Korea, in 1991. He is currently a Professor in the Department of Computer Engineering, Hongik University. He is interested in database technology, data science, big data processing and web technology.